

INDEXING BASISDATA CITRA DENGAN METODE KUANTISASI VEKTOR MENGGUNAKAN ALGORITMA FARI SHARE AMOUNT

Ary Mazharuddin S. – Febriliyan Samopa – Darlis Heru Murti

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember,
Email: shidiq@inf.its-sby.edu

ABSTRAK

Basisdata merupakan sesuatu yang selalu dipakai pada penyimpanan data-data baik dalam jumlah sedikit maupun banyak. Alasan penyimpanan dalam basisdata adalah keamanan data dan kemudahan dalam pencarian jika akan digunakan. Seiring berjalannya waktu, basisdata berkembang semakin kompleks. Kompleksitas tersebut dalam hal konsep maupun ragam data yang mampu disimpan di dalamnya. Jika dulu data yang dapat disimpan hanya berupa tulisan dan angka, maka sekarang basisdata tidak hanya mampu menyimpan tulisan dan angka, tapi juga gambar. Karena tidak semua data dapat dicari dengan metode yang sama, maka diperlukan metode pencarian data sesuai dengan tipe data yang dimaksud. Pada Tugas Akhir ini dikembangkan metode pencarian yang bisa diterapkan pada pencarian gambar dalam basisdata.

Metodologi yang digunakan dalam pembuatan Tugas Akhir ini terdiri dari beberapa tahapan. Pertama dilakukan pembuatan codebook sebagai acuan untuk melakukan indexing basisdata. Codebook ini dihasilkan dari algoritma Fair Share Amount (FSA). Tahapan selanjutnya adalah melakukan penyimpanan data dengan menggunakan acuan codebook yang sudah dibuat. Untuk mencari gambar, dilakukan dengan memasukkan parameter pencarian berupa gambar ke aplikasi dan mencocokkan parameter pencarian yang sudah terindeks dengan menggunakan codebook yang sama dengan yang digunakan pada penyimpanan.

Uji coba dan evaluasi dilakukan dengan menggunakan sejumlah gambar yang diambil secara acak dari yang sudah ada dalam basisdata untuk menguji akurasi. Pilihan yang digunakan untuk mengatur akurasi pencarian adalah tepat sama (kecocokan 100%) dan kesamaan (kecocokan 50%-90%). Selain itu juga digunakan gambar yang tidak ada dalam basisdata untuk menguji validitas pencarian. Data yang tersimpan dalam basisdata sejumlah 50 data. Hasil uji coba menunjukkan bahwa yang berpengaruh pada pencarian adalah ukuran gambar dan codebook yang digunakan, sedangkan warna dan pola gambar tidak berpengaruh pada pencarian

Kata Kunci : Kuantisasi Vektor, Fair Share Amount.

1. PENDAHULUAN

Dahulu data disimpan dalam bentuk fisik, seperti pada kertas-kertas berkas maupun pada benda-benda yang tahan dalam jangka waktu lama. Akan tetapi kendala akan ditemui manakala data yang mempunyai jumlahnya sangat banyak. Maka dikembangkan sebuah konsep penyimpanan yang disebut sebagai basisdata.

Ketika data tersimpan dalam basisdata, yang diharapkan adalah data tersebut dapat dengan aman tersimpan dan yang tidak kalah penting adalah ketika dibutuhkan, dapat dengan mudah diperoleh data yang diperlukan. Jika data yang tersimpan dalam jumlah sedikit maka tidak akan terlalu sulit bagi untuk menemukannya dan kemudian menggunakan sesuai dengan keperluan.

Akan tetapi yang perlu diperhatikan adalah jika basisdata yang tersimpan mempunyai skala yang cukup besar dan beragam jenisnya, misalkan data berbentuk image / citra. Karena tidak semua data dapat dicari dengan metode yang sama. Untuk itu

diperlukan metode pencarian data sesuai dengan tipe data yang dimaksud.

2. KUANTISASI VEKTOR

Metode kuantisasi vektor pertama kali dikembangkan pada tahun 1980, ketika Y. Linde, A. Buzo dan R. M Gray menulis An Algorithm for Vektor Quantizer Design, sehingga dapat dikatakan bahwa algoritma Linde-Buzo-Gray (LBG di kalangan komunitas VQ) merupakan algoritma kuantisasi vektor yang pertama. Sebelumnya kuantisasi vektor merupakan masalah yang rumit, karena dalam prosesnya melibatkan penyelesaian permasalahan multi dimensi. Dengan adanya algoritma LBG yang berdasarkan pada training vektor, maka metode kuantisasi vektor dapat berkembang sampai sekarang. Metode ini banyak digunakan dalam aplikasi kompresi gambar.

3. FAIR SHARE AMOUNT (FSA)

Ide dasar dari metode ini adalah sistem pemilu di Indonesia dimana setiap partai peserta pemilu

yang memperoleh suara di atas batas perolehan suara minimal akan memperoleh jatah perwakilan di DPR(D) sesuai dengan persentase perolehan suaranya. Juga pernyataan Y. Linde bahwa untuk data citra, training set yang evenly distributed cenderung memberikan hasil lebih baik. Karena itu metode ini bekerja dengan cara yang sama, yaitu dengan membagi vektor-vektor yang ada menjadi sejumlah area yang sama, dan kemudian menghitung perolehan “suara” setiap area untuk menentukan jumlah vektor perwakilan dari setiap area tersebut. Vektor-vektor perwakilan dari setiap area ini (yang jumlahnya masing-masing sesuai dengan persentase perolehan “suara”-nya) akan menjadi elemen dari codebook.

Dengan cara ini codebook dapat dihasilkan tanpa membutuhkan training set, sehingga proses training set dapat dihilangkan yang juga berarti pemangkasan waktu komputasi.

Cara kerja metode Fair-Share Amount untuk menghasilkan codebook dengan m jumlah elemen dari n buah vektor V adalah sebagai berikut :

1. Hitung nilai besaran untuk setiap vektor.
2. Urutkan vektor menaik berdasarkan nilai besarannya menggunakan Heap Sort.
3. Hitung besarnya area

$$SR = (\max(V) - \min(V)) / m$$
4. Bagi V menjadi m buah area (range) R_i yang masing-masing besarnya SR.
5. Hitung nilai tengah (median) dari setiap area R_i .
6. Hitung jumlah anggota A_i dari setiap R_i .
7. Cari pivot point (elemen yang memiliki nilai besaran yang paling mendekati tetapi tidak lebih besar dari median area tersebut) P_i untuk setiap R_i .
8. Hitung bobot W_i untuk setiap R_i dimana

$$W_i = \text{floor}((A_i / n) \times m)$$
9. Hitung elemen yang tersisa dengan rumus:

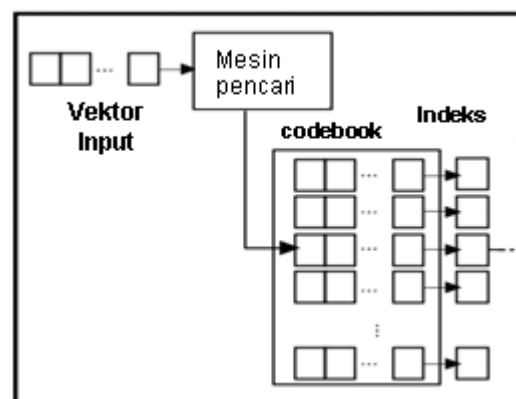
$$Z = m - W_i$$
10. Jika $Z > 0$, maka distribusikan Z ke setiap bobot W_i mulai dari W_i yang memiliki nilai terbesar sampai yang terkecil.
11. Ambil anggota R_i mulai dari elemen ke P_i – $\text{floor}(W_i / 2)$ sejumlah W_i untuk dijadikan sebagai elemen codebook.

4. DESKRIPSI SISTEM

Ada dua sistem yang menjadi kunci dari sistem yang dibangun ini, yaitu pembuatan codebook dan penyimpanan / pencarian.

4.1. PEMBUATAN CODEBOOK

Pembuatan codebook dilakukan dengan menggunakan data training sejumlah 100 gambar dengan ukuran 100x300 piksel dengan mode grayscale. Diagram pembuatan codebook ini dapat dilihat seperti gambar 1.



Gambar 1. pembuatan codebook

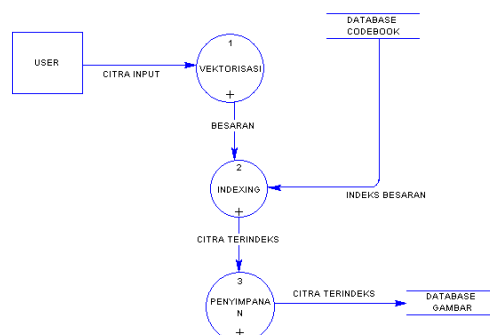
4.2. PENYIMPANAN/PENCARIAN

Proses penyimpanan dilakukan dengan vektorisasi gambar yang akan disimpan dalam basisdata, kemudian dilakukan penyimpanan gambar ke dalam tabel penyimpanan gambar beserta dengan indeksnya. Sedangkan pada proses pencarian, kita memberikan input berupa gambar yang akan kita cari dalam basisdata. Gambar tersebut akan divektorisasi sampai dengan menghasilkan urutan indeks besaran yang akan digunakan untuk pencarian gambar dalam basisdata. Selanjutnya setelah dihasilkan urutan indeks–indeks, maka dilakukan proses pencocokan dengan indeks yang dimiliki oleh gambar yang ada dalam basisdata. Dari sini akan diketahui jika terdapat gambar yang mempunyai kecocokan dengan inputan yang kita masukkan, maka akan menampilkan nama dari data yang dimaksud.

Proses penyimpanan terdiri dari:

1. Vektorisasi dan perhitungan besaran
2. Indexing atau pembacaan / pencocokan besaran dengan codebook yang tersedia (codebook 4 bit (16 codeword), 8 bit (256 codeword) dan 12 bit (4096 codeword))
3. Penyimpanan ke dalam basisdata.

Alur proses penyimpanan dapat dilihat pada gambar 2.



Gambar 2. Proses Penyimpanan data

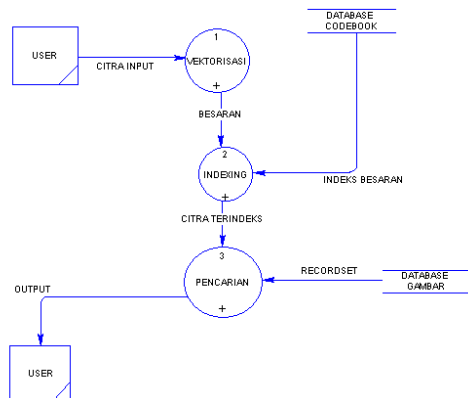
Proses pencarian terdiri dari:

1. Vektorisasi dan perhitungan besaran
2. Indexing atau pembacaan/pencocokan besaran dengan codebook yang tersedia (codebook 4

bit (16 codeword), 8 bit (256 codeword) dan 12 bit (4096 codeword))

3. Pencocokan indeks dari gambar yang ada dalam basisdata tersebut dengan indeks yang dihasilkan dari vektorisasi gambar yang digunakan sebagai parameter pencarian terhadap tiap baris data dalam tabel penyimpanan gambar.
4. Penampilan hasil pencarian dalam bentuk list nama gambar.

Alur proses pencarian dapat dilihat pada gambar 3



Gambar 3. Proses Pencarain

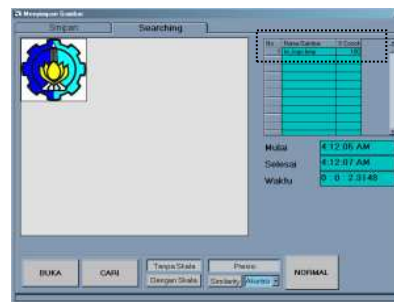
5. UJI COBA

Sistem yang telah diimplemetasikan akan diuji coba pada tahap ini untuk mengetahui apakah proses-proses yang telah dirancang dapat berjalan dengan baik. Ujicoba dilakukan dengan menggunakan skenario yang telah ditentukan dimana skenario ini berlaku untuk tiap jenis pencarian.

Uji coba dilakukan dengan menggunakan gambar sejumlah 50 buah yang tersimpan dalam basisdata. Ukuran gambar yang dipakai adalah 400 x 300 pixel dan 200 x 150 pixel. Pencarian dilakukan dengan menggunakan gambar yang ada dalam basisdata diambil secara acak untuk menguji apakah pencarian berjalan sebagaimana mestinya. Dan juga dilakukan pengujian terhadap akurasi pencarian dengan menggunakan pilihan-pilihan pencarian yang ada dalam aplikasi. Selain itu juga digunakan gambar yang tidak ada dalam basisdata untuk menguji validitas hasil pencarian.

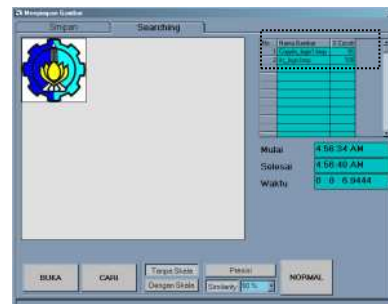
Pada uji coba ini diberikan beberapa skenario yang ditujukan untuk mengetahui fungsionalitas dari perangkat lunak yang dibuat. Terdapat tiga skenario yang diujikan pada metode hierarki dan non hierarki, pada masing-masing adalah:

- Pencarian pada gambar tanpa skala tepat sama. Contoh hasil pencarian dengan mode ini adalah seperti tampak pada gambar 4



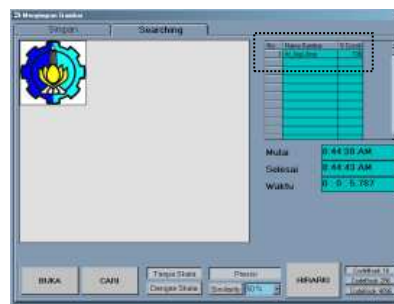
Gambar 4. Pencarian dengan mode tepat sama

Pencarian pada gambar tanpa skala dengan akurasi tertentu. Contoh hasil pencarian dengan mode ini adalah seperti tampak pada gambar 5



Gambar 5. Pencarian dengan akurasi tertentu tanpa skala

Pencarian dengan menggunakan skala dan akurasi tertentu. Contoh hasil pencarian dengan mode ini adalah seperti tampak pada gambar 6



Gambar 6. Pencarian dengan akurasi tertentu dan berskala

Pelaksanaan Skenario untuk Pencarian tanpa skala Tepat Sama dilakukan dengan tujuan untuk menguji pencarian dengan menggunakan gambar yang sama persis dengan yang ada dalam basisdata.

Pada skenario pertama, pencarian pada gambar tanpa menggunakan skala, baik pada metode hierarki maupun non hierarki, sama-sama hanya menghasilkan gambar yang persis sama dengan parameter pencarian. Waktu yang diperlukan untuk pencarian mendekati sama. Persentase kecocokan tidak berubah.

Pada skenario ini, jika digunakan gambar yang tidak ada dalam basisdata maka tidak akan menghasilkan apa pun.

Pelaksanaan Skenario untuk Pencarian tanpa skala Kesamaan dengan tujuan untuk menguji pencarian dengan menggunakan gambarparameter pencarian yang ukurannya sama persis dengan yang ada dalam basisdata akan tetapi mempunyai perbedaan dalam hal isi gambar, semisal ada noise pada gambar parameter pencarian atau yang terdapat dalam basisdata.

Pada skenario kedua, baik pada metode hierarki maupun non hierarki keduanya dapat menghasilkan gambar yang sama ukurannya dengan parameter pencarian dan melampaui batas minimum kecocokan yang ditentukan oleh pengguna aplikasi. Waktu yang diperlukan metode hierarki untuk menghasilkan gambar lebih lama daripada non hierarki dikarenakan proses dilakukan langsung berlapis sejumlah codebook. jika ditemukan lebih dari satu data setelah disaring dengan sebuah codebook. Proses tersebut hanya akan berhenti jika codebook sudah terpakai semua atau hanya ditemukan sebuah data pada sebuah lapisan codebook. Semakin kecil nilai kesamaan, maka akan semakin menambah waktu yang diperlukan untuk pencarian karena semakin besar peluang menghasilkan data yang jumlahnya lebih dari satu pada tiap lapisan.

Akan tetapi pada opsi pencarian ini tidak terlalu mencolok perbedaannya karena ukuran yang digunakan sama dengan yang tersimpan dalam basisdata. Pada metode non hierarki, persentase kecocokan cenderung menurun pada gambar yang kemiripannya makin sedikit seiring dengan semakin besarnya codebook yang digunakan.

Pada skenario ini, jika digunakan gambar yang tidak ada dalam basisdata maka tidak akan menghasilkan apa pun.

Pelaksanaan Skenario untuk Pencarian dengan skala Kesamaan dengan tujuan untuk menguji pencarian dengan menggunakan gambar parameter pencarian yang ukurannya bisa sama atau berbeda dengan yang ada dalam basisdata. Atau juga mempunyai perbedaan dalam hal isi gambar.

Pada skenario ketiga, metode pencarian hierarki akan menampilkan hasil yang mendekati 100 % kecocokan dengan parameter pencariin. Seperti halnya skenario kedua, waktu yang diperlukan relatif lebih lama karena menggunakan codebook berlapis.

Tabel 1. Pencarian Tanpa Skala Tepat Sama Hierarki

No	Nama Gambar	Ditemukan	Waktu(detik)
1	BATIGOLD1.bmp	1	41
2	BECK02.bmp	1	31
3	BECK05.bmp	1	36
4	BECK06.bmp	1	37
5	BECK07.bmp	1	37
6	BECK08.bmp	1	34
7	BECK09.bmp	1	38
8	BECK010.bmp	1	39
9	RUUD08.bmp	1	34
10	RUUD09.bmp	1	38
11	RUUD010.bmp	1	39
12	SCHOLES01.bmp	1	35
13	SCHOLES02.bmp	1	39
14	SCHOLES03.bmp	1	39
15	GIGGS02.bmp	1	35
16	SOLSKJAER03.bmp	1	37
17	SOLSKJAER04.bmp	1	41
18	ZIDANE01.bmp	1	38
19	ZIDANE02.bmp	1	38
20	ZIDANE03.bmp	1	38

Tabel 2. Pencarian Tanpa Skala Tepat Sama Non Hierarki

No	Nama Gambar	Ditemukan	Waktu(detik)
1	BATIGOLD1.bmp	1	38
2	BECK02.bmp	1	35
3	BECK05.bmp	1	38
4	BECK06.bmp	1	35
5	BECK07.bmp	1	37
6	BECK08.bmp	1	34
7	BECK09.bmp	1	34
8	BECK010.bmp	1	33
9	RUUD08.bmp	1	33
10	RUUD09.bmp	1	38
11	RUUD010.bmp	1	39
12	SCHOLES01.bmp	1	34
13	SCHOLES02.bmp	1	39
14	SCHOLES03.bmp	1	33
15	GIGGS02.bmp	1	35
16	SOLSKJAER03.bmp	1	37
17	SOLSKJAER04.bmp	1	37
18	ZIDANE01.bmp	1	38
19	ZIDANE02.bmp	1	38
20	ZIDANE03.bmp	1	38

Semakin kecil nilai kesamaan, maka akan semakin menambah waktu yang diperlukan untuk pencarian karena semakin besar peluang menghasilkan data yang jumlahnya lebih dari satu pada tiap lapisan. Pada metode hierarki, waktu yang diperlukan lebih lama daripada metode non hierarki karena menggunakan codebook berlapis.

Pada opsi pencarian ini akan sangat terlihat jika digunakan gambar dengan ukuran yang tidak sama persis dengan data yang terdapat dalam basisdata. Pada metode non hierarki hasil pencarian dan persentase kecocokan cenderung semakin meningkat seiring dengan makin besarnya codebook yang digunakan.

Tabel 3. Pencarian Tanpa Skala Kesamaan Hierarki

No	Nama Gambar	50% Kesamaan			90% Kesamaan		
		Ditemukan	Akurasi (%)	Waktu (detik)	Ditemukan	Akurasi (%)	Waktu (detik)
1	BATIGOLD1.bmp	1	100	43	1	100	40
2	BECK02.bmp	1	100	42	1	100	43
3	BECK05.bmp	1	100	39	1	100	38
4	BECK06.bmp	1	100	39	1	100	38
5	BECK07.bmp	1	100	42	1	100	42
6	BECK08.bmp	1	100	42	1	100	35
7	BECK09.bmp	1	100	43	1	100	38
8	BECK10.bmp	1	100	42	1	100	43
9	RUUD08.bmp	1	100	45	1	100	42
10	RUUD09.bmp	1	100	42	1	100	42
11	RUUD10.bmp	1	100	41	1	100	42
12	SCHOLES01.bmp	1	100	41	1	100	42
13	SCHOLES02.bmp	1	100	38	1	100	38
14	SCHOLES03.bmp	1	100	42	1	100	42
15	GIGGS02.bmp	1	100	41	1	100	42
16	SOLSKJAER03.bmp	1	100	41	1	100	42
17	SOLSKJAER04.bmp	1	100	41	1	100	43
18	ZIDANE01.bmp	1	100	24	1	100	38
19	ZIDANE02.bmp	1	100	42	1	100	39
20	ZIDANE03.bmp	1	100	39	1	100	42

Tabel 4. Pencarian Tanpa Skala Kesamaan non Hierarki

No	Nama Gambar	50% Kesamaan			90% Kesamaan		
		Ditemukan	Akurasi (%)	Waktu (detik)	Ditemukan	Akurasi (%)	Waktu (detik)
1	BATIGOLD1.bmp	3	50-100	39	1	100	41
2	BECK02.bmp	1	100	36	1	100	40
3	BECK05.bmp	1	100	40	1	100	36
4	BECK06.bmp	1	100	39	1	100	35
5	BECK07.bmp	3	100	38	1	100	41
6	BECK08.bmp	3	50-100	39	1	100	37
7	BECK09.bmp	1	100	40	1	100	36
8	BECK10.bmp	1	100	38	1	100	42
9	RUUD08.bmp	16	50-100	41	1	100	42
10	RUUD09.bmp	2	50-100	38	1	100	40
11	RUUD10.bmp	1	100	36	1	100	41
12	SCHOLES01.bmp	1	100	43	1	100	41
13	SCHOLES02.bmp	1	100	38	1	100	34
14	SCHOLES03.bmp	1	100	41	1	100	40
15	GIGGS02.bmp	1	100	43	1	100	43
16	SOLSKJAER03.bmp	1	100	41	1	100	35
17	SOLSKJAER04.bmp	1	100	40	1	100	40
18	ZIDANE01.bmp	1	100	39	1	100	41
19	ZIDANE02.bmp	3	50-100	39	1	100	39
20	ZIDANE03.bmp	5	50-100	39	1	100	37

Pada setiap skenario yang dipakai terdapat anomali hasil percobaan yang disebabkan karena metode pencarian yang dilakukan. Jika gambar yang dicari mempunyai banyak pixel yang berada di bawah batas bawah atau di atas batas atas codebook maka akan secara otomatis diberi nilai indeks batas bawah codebook atau batas atas codebook sehingga mereduksi pencarian.

Tabel 5. Pencarian dengan Skala Kesamaan Hierarki

No	Nama Gambar	50% Kesamaan			90% Kesamaan		
		Ditemukan	Akurasi (%)	Waktu (m.s)	Ditemukan	Akurasi (%)	Waktu (m.s)
1	BATIGOLD1.bmp	1	100	8.48	1	100	6.57
2	BECK02.bmp	1	100	11.83	1	100	5.57
3	BECK05.bmp	1	100	6.12	1	100	6.28
4	BECK06.bmp	1	100	12.15	1	100	6.53
5	BECK07.bmp	1	100	10.16	1	100	6.45
6	BECK08.bmp	1	100	7.48	1	100	5.57
7	BECK09.bmp	1	100	11.27	1	100	5.85
8	BECK10.bmp	1	100	8.23	1	100	6.28
9	RUUD08.bmp	1	100	12.5	1	100	5.53
10	RUUD09.bmp	1	100	10.25	1	100	6.45
11	RUUD10.bmp	1	100	8.9	1	100	6.57
12	SCHOLES01.bmp	1	100	11.77	1	100	5.68
13	SCHOLES02.bmp	1	100	6.75	1	100	6.28
14	SCHOLES03.bmp	1	100	7.6	1	100	6.52
15	GIGGS02.bmp	1	100	10.28	1	100	6.22
16	SOLSKJAER03.bmp	1	100	9.33	1	100	6.55
17	SOLSKJAER04.bmp	1	100	10.62	1	100	5.57
18	ZIDANE01.bmp	1	100	7.83	1	100	6.2
19	ZIDANE02.bmp	1	100	8.63	1	100	6.52
20	ZIDANE03.bmp	1	100	10.53	1	100	6.32

Tabel 6. Pencarian dengan Skala Kesamaan Hierarki

No	Nama Gambar	50% Kesamaan			90% Kesamaan		
		Ditemukan	Akurasi (%)	Waktu (m.s)	Ditemukan	Akurasi (%)	Waktu (m.s)
1	BATIGOLD1.bmp	13	50-100	5.55	1	100	5.75
2	BECK02.bmp	29	50-100	6.2	1	100	6.68
3	BECK05.bmp	20	50-100	5.33	1	100	5.35
4	BECK06.bmp	23	50-100	6.16	1	100	6.67
5	BECK07.bmp	25	50-100	5.4	1	100	5.77
6	BECK08.bmp	26	50-100	6.38	1	100	6.58
7	BECK09.bmp	21	50-100	5.11	1	100	5.91
8	BECK10.bmp	15	50-100	5.21	1	100	5.38
9	RUUD08.bmp	29	50-100	6.58	1	100	6.51
10	RUUD09.bmp	24	50-100	6.8	1	100	6.72
11	RUUD10.bmp	25	50-100	6.23	1	100	6.33
12	SCHOLES01.bmp	27	50-100	6.28	1	100	6.57
13	SCHOLES02.bmp	29	50-100	5.83	1	100	5.28
14	SCHOLES03.bmp	21	50-100	5.62	1	100	5.57
15	GIGGS02.bmp	22	50-100	5.46	1	100	5.4
16	SOLSKJAER03.bmp	23	50-100	6.76	1	100	6.3
17	SOLSKJAER04.bmp	21	50-100	6.41	1	100	6.8
18	ZIDANE01.bmp	20	50-100	5.91	1	100	5.62
19	ZIDANE02.bmp	19	50-100	5.16	1	100	5.52
20	ZIDANE03.bmp	18	50-100	6.95	1	100	6.43

Percobaan dengan menggunakan gambar yang tidak ada dalam basisdata, tidak akan menghasilkan keluaran yang diharapkan. Dengan menggunakan Codebook yang kecil dan kesamaan kecil, kemungkinan masih terdapat keluaran walaupun kecocokannya minim. Akan tetapi apabila besar codebook ditingkatkan atau nilai kesamaan ditingkatkan, maka keluaran yang tidak sesuai yang dihasilkan dari codebook dan kesamaan kecil tidak akan muncul.

6. KESIMPULAN

Beberapa kesimpulan yang dapat diambil dari Tugas Akhir ini adalah:

- Aplikasi ini mampu melakukan pencarian bertingkat dengan menggunakan codebook-codebook yang dimilikinya. Semakin besar codebook yang digunakan maka hasil pencarian akan semakin mendekati pada parameter pencarian.
- Kemampuan menentukan akurasi pencarian dapat diatur oleh pengguna. Jika ingin mendapatkan hasil yang paling mendekati parameter pencarian maka langkah yang diambil yang pertama adalah dengan memperbesar codebook yang digunakan, yang kedua adalah dengan meninggikan akurasi yang harus dipenuhi hasil pencarian dan yang ketiga adalah kombinasi dari keduanya.
- Codebook yang dihasilkan dari algoritma FSA ini berjalan dengan baik digunakan untuk melakukan indexing pada basisdata citra. Terbukti dengan uji coba yang dilakukan apabila terdapat gambar yang terdapat dalam database dapat ditemukan pada setiap pencarian. Sedangkan gambar yang tidak ada tidak akan menghasilkan apa-apa. Kesimpulan ini tentunya dengan memperhatikan kesimpulan nomor 2.

Beberapa saran yang dapat penulis berikan untuk Tugas Akhir ini adalah :

- Aplikasi ini dapat dikembangkan ke basis web. Di internet, banyak terdapat gambar-gambar yang bisa jadi suatu saat kita membutuhkannya. Dengan aplikasi semacam ini yang berbasis web, maka dapat dengan mudah kita dapatkan gambar yang kita cari.
- Aplikasi ini dapat dikembangkan pada sistem informasi kepegawaian atau semacamnya yang menggunakan gambar sebagai salah satu bagian penting dalam pengelolaannya.

7. DAFTAR PUSTAKA

1. "ImageGear The Imaging Toolkit of Choice : ActiveX User's Guide", AccuSoft Corp., Westborough, USA, 2000.
2. Lin, Jyh-Han, Vitter, Jeffrey Scott, "Nearly Optimal Vector Quantization via Linear Programming", Brown University, USA.
3. Langsam, Yedidyah, Augenstein, Moshe J, Tenenbaum, Aaron M, "Data Structures Using C And C++ ", Prentice Hall, Upper Saddle River, New Jersey 07458, 1996
4. Ryan, Michael J, Arnold, John F, "The Lossless Compression of AVIRIS Images by Vector Quantization", IEEE Transaction on Geoscience and Remote Sensing, Vol. 35-3, Page 546-550, May, 1997.
5. "Vector Quantization of Images", Brigham Young University, <http://orca.cs.byu.edu/~cline/ideas/vquant.html>.
6. "Vector Quantization, Information Theory", Delft University Technology,
7. <http://www-it.et.tudelft.nl/~college/et10-38/hc/hc6/index.htm>.
8. www.data-compression.com, "Theory of Data Compression".